

DELAY REDUCTION OF HARDWARE IMPLEMENTATION  
OF THE MAXIMUM A POSTERIORI (MAP) METHOD

Field of the Invention

5           The present invention relates to a method and/or  
architecture for a maximum a posteriori decoder generally and, more  
particularly, to a fast maximum a posteriori decoder that may be  
suitable for use in a turbo decoder.

10           Background of the Invention

Conventional turbo decoders are created using two  
conventional maximum a posteriori (MAP) decoders. A conventional  
MAP decoder implements equations 1-4 as follows:

15           
$$\log_e \Lambda_k = \log_e \frac{\sum_{u_k=+1} p(u, y)}{\sum_{u_k=-1} p(u, y)} \quad \text{Eq. (1)}$$

$$= \frac{\sum_{u_k=+1} \alpha_k^{s'} \gamma_k^{s',s} \beta_{k+1}^s}{\sum_{u_k=-1} \alpha_k^{s'} \gamma_k^{s',s} \beta_{k+1}^s}$$

$$\alpha_{k+1}^s = \sum_{s'} \alpha_k^{s'} \gamma_k^{s',s} \quad \text{Eq. (2)}$$

$$\beta_k^{s'} = \sum_s \gamma_k^{s',s} \beta_{k+1}^s \quad \text{Eq. (3)}$$

$$\gamma_k^{s',s} = \exp \left[ \sum_i y_i s_i \right] \quad \text{Eq. (4)}$$

5 The variable  $\Lambda_k$  is a ratio of a probability that a  $k^{\text{th}}$  bit is +1 to  
a probability that the  $k$ th bit is -1, given an encoder input signal  
sequence  $u$  and a received signal sequence  $y$ , and  $\log_e \Lambda_k$  is a log  
likelihood ratio (LLR). The variable  $\alpha_k^{s'}$  is a forward state metric  
of state  $s'$ . The variable  $\beta_{k+1}^s$  is a reverse state metric of state  
10  $s$ . The state metrics  $\alpha$  and  $\beta$  are calculated recursively by the  
expressions shown in equations (2) and (3). The variable  $\gamma_k^{s',s}$  is a  
branch metric, a quantity proportional to a probability that a  
transition is made from state  $s'$  to state  $s$ . The variable  $y_i$  is an  
 $i^{\text{th}}$  received noisy code word having the state  $s_i$ .

15 In a conventional hardware implementation, the product  
 $\alpha_k^{s'} \gamma_k^{s',s} \beta_{k+1}^s$  of equation 1 is more efficiently implemented by a sum.  
The sum is accomplished by a transformation of the variables  $\alpha$ ,  $\beta$   
and  $\gamma$  by taking logarithms as shown in equations 5-7 as follows:

$$\hat{\alpha} = \log_e \alpha \quad \text{Eq. (5)}$$

$$\hat{\beta} = \log_e \beta \quad \text{Eq. (6)}$$

$$\hat{\gamma} = \log_e \gamma = \sum_i y_i s_i \quad \text{Eq. (7)}$$

5 The products in the summations in equation 1 become additions as shown in equation 8 as follows:

$$\begin{aligned} \log_e \Lambda_k &= \log_e \frac{\sum_{u_k=+1} e^{\hat{\alpha}_k^{s'}} e^{\hat{\gamma}_k^{s',s}} e^{\hat{\beta}_{k+1}^s}}{\sum_{u_k=-1} e^{\hat{\alpha}_k^{s'}} e^{\hat{\gamma}_k^{s',s}} e^{\hat{\beta}_{k+1}^s}} \quad \text{Eq. (8)} \\ &= \log_e \frac{\sum_{u_k=+1} \exp[\hat{\alpha}_k^{s'} + \hat{\gamma}_k^{s',s} + \hat{\beta}_{k+1}^s]}{\sum_{u_k=-1} \exp[\hat{\alpha}_k^{s'} + \hat{\gamma}_k^{s',s} + \hat{\beta}_{k+1}^s]} \end{aligned}$$

10 Rather than summing the products of the state metrics and branch metrics, calculation of the LLR becomes a summation of the exponentials of the sum of the state metrics and branch metrics. The summation can be computed using conversions shown in equations  
15 9 and 10 as follows:

$$\begin{aligned}\log_e(e^A + e^B) &= \max(A, B) + \log_e(1 + e^{-|A-B|}) \\ &= \max^*(A, B)\end{aligned}\quad \text{Eq. (9)}$$

$$\max^*(A, B, C) = \max^*(\max^*(A, B), C) \quad \text{Eq. (10)}$$

5

The maximum-log function ( $\max^*$ ) can be interpreted as a maximum function  $\max(A, B)$  with a correction factor  $\log_e(1 + e^{-|A-B|})$ . With the conversions, the MAP decoder equations can be expressed in terms of  $\max^*$  functions. The forward state metric equation in terms of the  $\max^*$  function is shown in equations 11 and 12, for illustration, as follows:

$$\alpha_{k+1}^s = \sum_{s'} \alpha_k^{s'} \gamma_k^{s',s} \quad \text{Eq. (11)}$$

$$\hat{\alpha}_{k+1}^s = \log_e \alpha_{k+1}^s = \log_e \sum_{s'} \alpha_k^{s'} \gamma_k^{s',s} \quad \text{Eq. (12)}$$

$$= \log_e \sum_{s'} \exp[\alpha_k^{s'} + \gamma_k^{s',s}]$$

$$= \max^* \left( \sum_{s'} (\hat{\alpha}_k^{s'} + \hat{\gamma}_k^{s',s}) \right)$$

Referring to FIG. 1, a conventional core circuit 10 of a conventional turbo decoder architecture is shown. A received bit

01-819  
1496.00190

signal (i.e.,  $Y$ ) and extrinsic information signal (i.e.,  $E$ ) are used to calculate branch metric signals (i.e.,  $\gamma$ ) in a branch metrics circuit 12. The branch metric signals  $\gamma$  are summed with state metric signals (i.e.,  $\alpha_i$  and  $\beta_j$ ) in a state metrics circuit 14 to determine the state metrics for a next state metric signal (i.e.,  $\alpha_{i+1}$  and  $\beta_{j+1}$ ). The sums (i.e., intermediate signals  $\alpha+\gamma$  and  $\beta+\gamma$ ) are also passed onto a log likelihood ratio (LLR) circuit 16. The LLR circuit 16 calculates multiple log likelihood ratios that determined the most likely data transitions. The LLR circuit 16 then presents the most likely data transitions in a data signal (i.e.,  $U$ ).

A bottleneck of the conventional turbo decoder core circuit 10 is the state metrics circuit 14. Because of the recursive nature of operations within the state metrics circuit 14, introduction of pipeline architecture cannot improve the overall throughput of the core circuit 10. Successive state metrics are calculated in a specific sequence.

Referring to FIG. 2, a detailed block diagram of the conventional state metrics circuit 14 is shown. The state metrics circuit 14 performs multiple operations, an addition 17, a maximum-

01-819  
1496.00190

log (MAX\*) operation 18, a maximum (MAX) operation 20, and a normalization (NORM) operation 22. Values of the forward state metric signals  $\hat{\alpha}_k$  and the branch metric signals  $\hat{\gamma}_k$  are summed 17 and the max\* function applied in the maximum-log operation 18 to  
5 generate a next forward state metric signal  $\hat{\alpha}'_{k+1}$ . Calculation of a next reverse state metric signal (not shown) is performed in a similar manner. The maximum operation 20 is then performed on the next forward state metric signal  $\hat{\alpha}'_{k+1}$  to generate a normalization signal (i.e., N).

10 The value of the next forward state metric signal  $\hat{\alpha}'_{k+1}$  needs to be normalized because the next forward state metric signal  $\hat{\alpha}'_{k+1}$  is representing numbers with a finite number of bits. All of the original forward state metric signals  $\alpha$  are also normalized to maintain a proper relative amplitude. The next forward state  
15 metric signal  $\hat{\alpha}'_{k+1}$  and the forward state metric signals  $\alpha$  are then normalized by a value of the normalization signal N in the normalization operation 22. Since the normalization operation 22 depends on the results of the maximum-log operation 18 and the maximum operation 20, the steps executed by the circuits 18, 20,  
20 and 22 are inherently sequential. For example, a conventional core

01-819  
1496.00190

circuit 10 fabricated by a conventional 0.25 micron technology resulted in a synthesized delay of 16.3 nanoseconds.

### Summary of the Invention

5           The present invention concerns a decoder generally comprising a branch metrics circuit and a state metrics circuit. The branch metrics circuit may be configured to generate a plurality of branch metric signals. The state metrics circuit may be configured to (i) add the branch metric signals to a plurality  
10 of state metric signals to generate a plurality of intermediate signals, (ii) determine a next state metric signal to the state metric signals, (iii) determine a normalization signal in response to the intermediate signals, and (iv) normalize the state metric signals in response to the normalization signal.

15           The objects, features and advantages of the present invention include providing a maximum a posteriori decoder that may (i) reduce a delay through a turbo decoder core circuit, (ii) occupy less space and/or (iii) require fewer decoders to meet a specific baud rate requirement.

20

**Brief Description of the Drawings**

These and other objects, features and advantages of the present invention will be apparent from the following detailed description and the appended claims and drawings in which:

5           FIG. 1 is a block diagram of a conventional core circuit for a turbo decoder circuit;

FIG. 2 is a block diagram illustrating an operation of the conventional core circuit;

FIG. 3 is a block diagram of a turbo decoder circuit;

10           FIGS. 4A-C are diagrams illustrating normalization methods; and

FIG. 5 is a block diagram of a core circuit according to the present invention.

15           **Detailed Description of the Preferred Embodiments**

Referring to FIG. 3, a block diagram of a turbo decoder circuit 100 is shown in accordance with a preferred embodiment of the present invention. The turbo decoder circuit 100 generally comprises a circuit 102, a circuit 104, a circuit 106, and a  
20           circuit 108. A signal (e.g., Y) may be received at an input 110 of the turbo decoder circuit 100 through a channel 112. Another



01-819

1496.00190

signal (e.g., Y') may be received at an input 114 of the turbo decoder circuit 100 through the channel 112. A signal (e.g., Y'') may be received at an input 116 of the turbo decoder circuit 100 through the channel 112. The turbo decoder circuit 100 may have an  
5 output 118 to present a signal (e.g., U).

The signal Y may represent data carried through the channel 114. The signal Y' may represent parity information for the signal Y. The signal Y'' may represent more parity information for the signal Y that has been interleaved with respect to the  
10 signal Y. The signals Y, Y' and Y'' may include noise induced by the channel 112. The signal U may represent the data extracted from the signals Y, Y' and Y'' as determined by the turbo decoder circuit 100.

The circuit 102 may be implemented as a maximum a  
15 posteriori (MAP) decoder circuit. The MAP decoder circuit 102 may receive the signal Y. The MAP decoder circuit 102 may also receive the signal Y'. The MAP decoder circuit 102 may present the signal U. A signal (e.g., E12) may be generated by the MAP decoder circuit 102 and presented to the circuit 106. A signal (e.g.,  
20 E21') may be received by the MAP decoder circuit 102 from the circuit 108.

01-819  
1496.00190

The circuit 104 may be implemented as another MAP decoder circuit. The MAP decoder circuit 104 may receive the signal Y. The MAP decoder circuit 104 may receive the signal Y". A signal (e.g., E12') may be received by the MAP decoder circuit 104 from the circuit 106. A signal (e.g., E21) may be generated by the MAP decoder circuit 104 and presented to the circuit 108.

The circuit 106 may be implemented as an interleave circuit. The interleave circuit 106 may interleave the signal E12 to generate and present the signal E12'. The signal E12 may be soft or extrinsic information generated by the MAP decoder circuit 102 for use by the MAP decoder circuit 104.

The circuit 108 may be implemented as a de-interleave circuit. The de-interleave circuit 108 may de-interleave the signal E21 to generate and present the signal E21'. The signal E21 may be soft or extrinsic information generated by the MAP decoder circuit 104 for use by the MAP decoder circuit 102. The MAP decoder circuits 102 and 104 may thus operate as iterative soft-output decoders within the turbo decoder circuit 100.

The channel 112 may be implemented as any type of channel that may convey data. For example, the channel 112 may include, but is not limited to, radio-frequency channels, fiber optic

channels, infrared communication channels, and the like. Other types of channels may be used with the present invention to meet the design criteria of a particular application. Multiple turbo decoder circuits 100 may be coupled to the channel 112 where the channel 112 has an ability to deliver information faster than a single turbo decoder circuit 100 can decode the signals Y, Y', and Y".

Referring to FIGS. 4A-C, diagrams illustrating normalization methods are shown. The FIG. 4A is a diagram illustrating a bit truncation type of normalization method. The bit truncation method is generally fast, but performance of the turbo decoder circuit 100 generally suffers. By way of example, a next forward state metric signal (e.g.,  $\alpha'$ ), represented by m bits and having a data range 120, may be truncated/normalized to generate a normalized forward state metric signal (e.g.,  $\alpha$ ) represented by m-1 bits. The truncation approach subtracts  $2^{m-1}$  from the next forward state metric signal  $\alpha'$ , assigning a zero (0) value to the normalized forward state metric signal  $\alpha$  if the subtraction results in a negative value.

While the bit truncation method may be simple and fast when implemented in hardware, the performance of the truncation

01-819  
1496.00190

method generally suffers due to increased bit error rate (BER) resulting from the loss in dynamic range of the normalized forward state metric signal  $\alpha$ . It may be apparent that the entire data range 120 may be completely represented by the (m-bit) next forward state metric signal  $\alpha'$ . In contrast, an effective data range 122A of the (m-1 bit) normalized forward state metric signal  $\alpha$  may be significantly lower than the data range 120. Furthermore, a large portion 124A of the next forward state metric signal  $\alpha'$  may be lost.

Referring to FIG. 4B, a diagram illustrating a maximum-range type normalization method is shown. The maximum-range normalization method may first determine a maximum value (e.g.,  $\alpha_{\max}$ ) of the next forward state metric signal  $\alpha'$ . An amount (e.g.,  $\alpha_{\max} - 2^{m-1}$ ) may then be subtracted from the normalized and all other forward state metric signals  $\alpha$  of a data set, again assigning a zero (0) value if the subtraction results in a negative value.

The maximum-range approach may use a maximum dynamic range offered by the normalized forward state metric signal  $\alpha$ , at the expense of a delay because the max\* function calculation and the maximum-range normalization calculation may be performed sequentially. The maximum-range method may subtract a smaller

01-819

1496.00190

amount from the original next forward state metric signal  $\alpha'$  than the bit truncation method. As a result, an effective data range 122B of the normalized forward state metric signal  $\alpha$  may be larger for the maximum-range method than the effective data range 122A for the bit truncation method. Likewise, a portion 124B of the next forward state metric signal  $\alpha'$  lost by the maximum-range method may be smaller than the portion 124A lost by the bit truncation method.

Referring to FIG. 4C, a diagram illustrating a parallelization type of normalization method is shown. A similarity of the  $\max^*$  function to the maximum function generally allows the performance of the turbo decoder circuit 100 to be retained and the hardware delay reduced at the same time. The parallelization method may normalize by a maximum value determined by  $(\alpha + \gamma)$  rather than a maximum value determined by  $\alpha'$  or  $\max^*(\alpha + \gamma)$ . The normalization operation and  $\max^*$  function computation may thus be performed in parallel simultaneously.

A general difference between the parallelization approach and the maximum-range method is that the maximum function neglects the correction factor of the  $\max^*$  function. To account for the correction factor, the parallelization method may further adjusts

01-819

1496.00190

the normalization operation by a guard band value (e.g., GB) to  
determined the normalized forward state metric signal  $\alpha$ . The guard  
band value GB may be determined by the correction factor of the  
max\* function. In the example shown in FIG. 4C, the normalized  
5 forward state metric signal  $\alpha$  may be represented by seven (7) bits  
for a maximum possible value (e.g., MV) of 127 and the determined  
guard band value GB may have a value twelve (12). An effective  
data range 122C of the normalized forward state metric signal  $\alpha$  may  
be slightly smaller for the parallelization method than the  
10 effective data range 122B for the maximum-range method. A lost  
portion 124C of the original next forward state metric signal  $\alpha'$   
may be slightly larger for the parallelization method than the lost  
portion 124B for the maximum-range method.

Simulations of bit error rates (BER) as a function of a  
15 signal to noise ratio (SNR) for the maximum-range and the  
parallelization normalization methods have been performed for  
comparison purposes. The simulations generally show that the BER  
performance for a turbo decoder are equally good for the two  
methods. In general, the BER does not change when implementing the  
20 maximum-range normalization method or the parallelization  
normalization method.

Referring to FIG. 5, a block diagram of a circuit 138 implementing a preferred embodiment of the present invention is shown. The circuit 138 may represent a core circuit in each of the MAP decoder circuits 102 and 104 of the turbo decoder 100. The  
5 core circuit 138 generally comprises a circuit 140, a circuit 142, and a circuit 144.

The circuit 140 may be implemented as a branch metrics circuit. The branch metrics circuit 140 may receive several signals (e.g.,  $Y$ ,  $Y'$ ,  $Y''$  and  $E$ ). The branch metrics circuit 140  
10 may operate similar to the branch metrics circuit 12 to generate and present multiple branch metric signals (e.g.,  $\gamma$ ) in response to the signals  $Y$ ,  $Y'$  or  $Y''$  and  $E$ . The branch metric signals  $\gamma$  may be implemented as a fixed-point variables in a hardware implementation.

15 The circuit 142 may be implemented as a log likelihood ratio (LLR) circuit. The LLR circuit 142 may receive the state metric signals  $\alpha$  and  $\beta$ . The LLR circuit 142 may also receive multiple intermediate signals (e.g.,  $\alpha+\gamma$  and  $\beta+\gamma$ ). The LLR circuit 142 may operate similar to the LLR circuit 16 to generate and  
20 present the data signal  $U$  in response to (i) the state metric

01-819  
1496.00190

signals  $\alpha$  and  $\beta$  and (ii) the intermediate signals  $\alpha+\gamma$  and  $\beta+\gamma$ . The state metric signals  $\alpha$  and  $\beta$  may be implemented as fixed-point variables in the hardware. The intermediate signals  $\alpha+\gamma$  and  $\beta+\gamma$  may also be implemented as fixed-point variables in the hardware.

5           The circuit 144 may be implemented as a state metrics circuit. The state metrics circuit 144 may receive the state metric signals  $\alpha$  and  $\beta$ . The state metrics circuit 144 may also receive the branch metric signals  $\hat{\gamma}_k$ . The state metrics circuit 144 may iterate calculations of the state metric signals  $\alpha$  and  $\beta$  and the intermediate signals  $\alpha+\gamma$  and  $\beta+\gamma$  to determine normalized and then most likely values. The state metric signals  $\alpha$  and  $\beta$  presented by the state metrics circuit 144 may be stored in a memory circuit (not shown). The state metric signals  $\alpha$  and  $\beta$  may be implemented as fixed-point variables in the hardware.

15           The state metrics circuit 144 generally comprises a circuit 146, a circuit 148, a circuit 150 and a circuit 152 to determine the forward state metric signals  $\alpha$ . Similar circuits (not shown) may be included in the state metrics circuit 144 to determine the backward state metric signals  $\beta$ . Each of the  
20           circuits 146, 148, 150 and 152 may be implemented with fixed-point



variables. Likewise, each signal received, generated, and/or presented by each of the circuits 146, 148, 150 and 152 may be represented by fixed-point variables.

The circuit 146 may be implemented as an adding circuit.

5 The adding circuit 146 may receive the forward state metric signals  $\hat{\alpha}_k$ . The adding circuit 146 may also receive the branch metric signals  $\hat{\gamma}_k$ . The adding circuit 146 may add the forward state metric signals  $\hat{\alpha}_k$  and the branch metric signals  $\hat{\gamma}_k$  to generate intermediate signals  $\hat{\alpha}_k + \hat{\gamma}_k$ . The intermediate signals  $\hat{\alpha}_k + \hat{\gamma}_k$  may  
10 be presented to the circuit 148, the circuit 150 and the LLR circuit 142 simultaneously.

The circuit 148 may be implemented as a maximum-log circuit. The circuit 148 may receive the intermediate signals  $\hat{\alpha}_k + \hat{\gamma}_k$ . The circuit 148 may be configured to perform the maximum-  
15 log (max\*) operation on the intermediate signals  $\hat{\alpha}_k + \hat{\gamma}_k$  to generate the next forward state metric signal  $\hat{\alpha}'_{k+1}$ . The next forward state metric signal  $\hat{\alpha}'_{k+1}$  and all other existing forward state metric signals  $\alpha$  may be presented to the circuit 152 for normalization to avoid an overflow of the fixed-point variables.

20 The circuit 150 may be implemented as a maximum circuit. The maximum circuit 150 may be configured to perform the maximum

operation on the intermediate signals  $\hat{\alpha}_k + \hat{\gamma}_k$  to generate a signal (e.g., N). The signal N may be implemented as a normalization factor signal. The signal normalization factor N may be presented to the circuit 152. The normalization factor signal N may be generated independently of the correction factor used in the maximum-log circuit 148. Therefore, the maximum circuit 150 may operate simultaneously with the maximum-log circuit 148. The ability of the maximum-log circuit 148 and the maximum circuit 150 to operate in parallel generally allows the state metrics circuit 144 to iterate the state metric signals  $\alpha$  and  $\beta$  with a shorter delay than the conventional state metrics circuit 14.

The circuit 152 may be implemented as a normalization circuit. The normalization circuit 152 may simultaneously receive the next forward state metric signal  $\hat{\alpha}_{k+1}$  from the maximum-log circuit 148, the normalization factor signal N from the maximum circuit 150, and the existing forward state metric signals  $\alpha$ . The normalization circuit 152 may be configured to normalize the next forward state metric signal  $\hat{\alpha}_{k+1}$  and the forward state metric signals  $\alpha$  to generate an iterated set of the state metric signals  $\alpha$ . The parallelization normalization method may be implemented to generate the iterated set of the state metric signals  $\alpha$ . The state

01-819  
1496.00190

metric signals  $\alpha$  may then be stored in the memory circuit (not shown) and then reiterated multiple times using the extrinsic information from the signals E12' and E21' to improve a probability of detecting the data within the signal Y. Other designs of the normalization circuit 152 may be implemented to meet the design criteria of a particular application.

Including a compiler optimized maximum-log (max\*) function and the parallelization, an example delay of the turbo decoder core circuit 138 may be reduced to approximately 9.1 nanoseconds, which is generally 45% less than the original delay of 16.3 ns for the conventional core circuit 10. The increase in speed generally means that fewer turbo decoder circuits 100 may be needed to meet any specific baud rate requirement of a particular application. Performance examples of the conventional core circuit 10 and the core circuit 138 of the present invention are shown in Table I.

TABLE I

| Turbo Core circuit                           |              |           |
|--|--------------|-----------|
|  | Conventional | Optimized |
| Delay (ns)                                   | 16.3         | 9.1       |
| Overall                                      |              |           |
| No. of decoders at 21 Mbaud/s (6 iterations) | 6            | 4         |
| No. of decoders at 45 Mbaud/s (6 iterations) | 10           | 6         |
| No. of decoders at 21 Mbaud/s (8 iterations) | 7            | 5         |
| No. of decoders at 45 Mbaud/s (8 iterations) | 13           | 8         |

As used herein, the term "simultaneously" is meant to describe events that share some common time period but the term is not meant to be limited to events that begin at the same point in time, end at the same point in time, or have the same duration.

The various signals of the present invention may be implemented as single-bit or multi-bit signals in a serial and/or parallel configuration.

While the invention has been particularly shown and described with reference to the preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made without departing from the spirit and scope of the invention.